



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/733,973	12/11/2003	Richard D. Dettinger	ROC920030307US1	1355

7590 11/15/2006  
William J. McGinnis, Jr.  
IBM Corporation  
Dept. 917  
3605 Highway 52 North  
Rochester, MN 55901-7829

EXAMINER

PONIKIEWSKI, TOMASZ

ART UNIT PAPER NUMBER

2165

DATE MAILED: 11/15/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b> 10/733,973	<b>Applicant(s)</b> DETTINGER ET AL.	
	<b>Examiner</b> Tomasz Ponikiewski	<b>Art Unit</b> 2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

#### Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

#### Status

- 1) ☒ Responsive to communication(s) filed on 30 August 2006.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

#### Disposition of Claims

- 4) ☒ Claim(s) 1-12, 15-30 and 32-35 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-12, 15-30 and 32-35 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

#### Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

#### Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

#### Attachment(s)

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)                                | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                       | 5) <input type="checkbox"/> Notice of Informal Patent Application                       |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

### **DETAILED ACTION**

1. The Amendment filed on August 30, 2006 has been received and entered. Claims 13, 14 and 31 have been cancelled. Claim 35 has been added. Therefore, claims 1-12, 15-30 and 32-35 are pending.
2. The Applicant's communication overcomes some objections and rejections under 112.

### ***Claim Objections***

3. Claims 1, 10, 15, 24, 29, 30, 32 and 35 are objected to because of the following informalities:

Claims 1, 10, 15, 29, 32 and 35 are objected to because of the following informalities: the recitation "whereby" in the body of the claims should be changed to "includes" or simply deleting the recitation. As pointed out by the applicant MPEP 2111.04 recites that a phrase following "whereby" carries no weight when it expresses the intended result. Appropriate correction is required.

Claims 2, 4, 15 and 17-20 recite the word "for" in the body of the claims. It indicates intended use and as such does not carry patentable weight. The word could be changed to recite "to". The limitations following the phrase "for" describes only

Art Unit: 2165

intended use but not necessarily required functionality of the claim. Limitations following the phrase “for” do not carry patentable weight, which cause the claims to appear as a series of non-functional descriptive material/data without any functional relation with each other. Applicant is required to amend the claims so that the claim limitations are recited in a definite form. For example, claim 17 recites “for execution” should be “to execute” or “that executes”.

***Claim Rejections - 35 USC § 102***

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

5. Claims 1-12, 15-28 and 35 are rejected under 35 U.S.C. 102(e) as being anticipated by Li (US 6,748,386 B1).

As per claim 1 Li is directed to a computer-implemented method of execution of a multi-step workflow that is repeatedly executed on data of a database, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format, the method comprising:

receiving current input for execution of a step of the workflow on relevant data of the database, wherein the step has been previously executed on the relevant data using previous input identical to the current input and wherein the previous execution of the step produced previous output (column 6, lines 55-56; column 7, lines 55-61);

determining whether the step is deterministic, whereby the step generates identical output for given input in repeated executions of the step on the relevant data (column 7, lines 54-61, wherein the cached results would not be retrieved if the query wasn't the same);

and if the step is deterministic, returning the previous output produced during the previous execution of the step without re-executing the step (column 7, lines 55-61).

As per claim 2 Li is directed to comprising using the returned previous output as input to a next sequential step in the workflow (column 7, lines 54-61, wherein saved and retrieved once may mean it could be used again for next step in the process).

As per claim 3 Li is directed to determining whether the step is deterministic comprises determining whether a workflow description of the step includes a deterministic flag indicating that the step generates identical output for given input in repeated executions of the step on the relevant data (column 7, lines 54-61, wherein since the saved result is retrieved that means there is some sort of indicator like a flag to show that the output is the same).

Art Unit: 2165

As per claim 4 Li is directed to determining whether the current input and the previous input are the same (column 7, lines 57-61, wherein since the saved result is retrieved means that there was a process of determination);

and returning the previous output produced during the previous execution of the step only if the current input and the previous input are determined to be the same (column 7, lines 57-61).

As per claim 5 Li is directed to determining whether the current input and the previous input are the same comprises accessing a hash table representative of the previous input (column 7, lines 52-53, wherein has table could be cache).

As per claim 6 Li is directed to determining whether the relevant data has been changed since the previous execution (column 7, lines 25-28);

and returning the previous output produced during the previous execution of the step only if the relevant data has not been changed (column 7, lines 57-61).

As per claim 7 Li is directed to determining whether the relevant data has been changed comprises:

determining a timestamp indicating a point of time of the previous execution (column 7, lines 53-55);

and determining, from a transaction log of the database, whether transactions relative to the relevant data have occurred since the point of time indicated by the timestamp (column 7, lines 61-66).

As per claim 8 Li is directed to comprising:

if the relevant data has been changed since the previous execution:

executing the step on the relevant data to obtain a result (column 6, lines 55-57);

and storing the result as output to be returned for subsequent invocations of the step taking input identical to the current input, in which case execution of the step is avoided and the stored output is returned for the step (column 7, lines 7-9).

As per claim 9 Li is directed to wherein the current input comprises one or more result fields and input parameters (column 6, lines 13-14, wherein "input parameters" could be multiple "parameters").

As per claim 10 Li is directed to a computer-implemented method for managing execution of a workflow that is repeatedly executed on data of a database, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format, the method comprising:

Art Unit: 2165

receiving current input for execution of a step of the workflow on relevant data of the database, (column 7 lines 3-7);

identifying the step as deterministic, wherein the step generates identical output for given input in repeated executions of the step on the relevant data (column 7, lines 56-57, wherein if the result check of the query is cached then it means that another query with identical input has been executed prior to this instance)

upon determining that the step had been previously executed using input identical to the current input (column 7, lines 25-24):

returning output obtained in the previous execution of the step using input identical to the current input without executing the step using the received current input (column 7, lines 52-61); and

upon determining that the step has not been previously executed using input identical to the current input:

executing the step for the current input on the relevant data to obtain a result (column 7, lines 17-21); and

storing the result to enable managing a next invocation of the step in which the step is passed input identical to the current input, in which case the stored result is returned as output for the step without re-executing the step (column 7, lines 21-24).



As per claim 11 Li is directed to further comprising, upon determining that the step has been previously executed using input identical to the current input and prior to returning the output:

determining whether the relevant data has been changed since the previous execution of the step using the input identical to the current input (column 7, lines 25-28); and

if the relevant data has not been changed, retrieving the output obtained in the previous execution of the step using the input identical to the current input (column 7, lines 57-59).

As per claim 12 Li is directed to determining whether the relevant data has been changed comprises:

determining a timestamp indicating a point of time of the previous execution (column 7, lines 54-57);

and determining, from a transaction log of the database, whether transactions relative to the relevant data have occurred since the point of time indicated by the timestamp (column 7, lines 62-66).

As per claim 15 Li is directed to a computer readable storage medium containing a program which, when executed by a processor, performs an operation of managing execution of a multi-step workflow that is repeatedly executed on data of a database, wherein the workflow is defined by plurality of steps, each step being an executable

Art Unit: 2165

function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format, the operation comprising:

receiving current input for execution of a step of the workflow on relevant data of the database, wherein the step has been previously executed on the relevant data using previous input identical to the current input and wherein the previous execution of the step produced previous output (column 6, lines 55-56; column 7, lines 55-61);

determining whether the step is deterministic, whereby the step generates identical output for given input in repeated executions of the step on the relevant data (column 7, lines 54-61, wherein the cached results would not be retrieved if the query wasn't the same);

and if the step is deterministic, returning the previous output produced during the previous execution of the step without re-executing the step (column 7, lines 55-61).

As per claim 16 Li is directed to the operation further comprises: inputting the returned previous output to a next sequential step in the workflow (column 7, lines 55-61).

As per claim 17 Li is directed to determining whether the step is deterministic comprises determining whether a workflow description of the step includes a deterministic flag indicating that the step generates identical output for given input in repeated executions of the step on the relevant data (column 7, lines 54-61, wherein

Art Unit: 2165

since the saved result is retrieved that means there is some sort of indicator like a flag to show that the output is the same).

As per claim 18 Li is directed to the operation further comprises:

determining whether the current input and the previous input are the same (column 7, lines 57-61, wherein since the saved result is retrieved means that there was a process of determination);

and returning the previous output produced during the previous execution of the step only if the current input and the previous input are determined to be the same (column 7, lines 57-61).

As per claim 19 Li is directed to determining whether the current input and the previous input are the same comprises accessing a hash table representative of the previous input (column 7, lines 52-53, wherein has table could be cache).

As per claim 20 Li is directed to the operation further comprises:

determining whether the relevant data has been changed since the previous execution (column 7, lines 25-28);

and returning the previous output produced during the previous execution of the step only if the relevant data has not been changed (column 7, lines 57-61).

Art Unit: 2165

As per claim 21 Li is directed to determining whether the relevant data has been changed comprises:

retrieving a timestamp indicating a point of time of the previous execution (column 7, lines 53-55);

and retrieving a transaction log of the database (column 7, line 59, wherein the log is the query holding the query information);

and determining, from the transaction log, whether transactions relative to the relevant data have occurred since the point of time indicated by the timestamp (column 7, lines 61-66).

As per claim 22 Li is directed to the operation further comprises:

if the relevant data has been changed since the previous execution:

executing the step on the relevant data to obtain a result (column 6, lines 55-57);

and storing the result as the output to be returned for subsequent invocations of the step taking input identical to the current input, in which case execution of the step is avoided and the stored output is returned for the step (column 7, lines 7-9).

As per claim 23 Li is directed to the current input comprises one or more result fields and input parameters (column 6, lines 13-14, wherein "input parameters" could be multiple "parameters").

As per claim 24 Li is directed to a computer readable storage medium containing a program which, when executed by processor, performs an operation of managing execution of a workflow that is repeatedly executed on data of a database, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format, the operation comprising:

receiving current input for execution of a step of the workflow on relevant data of the database, wherein the step generates identical output for given input in repeated executions of the step on the relevant data (column 6, lines 64-67; column 7, lines 1-7);

and without executing the step using the current input, returning output obtained in a previous execution of the step using input identical to the current input (column 7, lines 52-61).

As per claim 25 Li is directed to the operation further comprises, prior to returning the output:

determining whether the step has been previously executed using the input identical to the current input (column 7, lines 3-7, wherein the wrapper can then check if identical query has been executed);

if so, determining whether the relevant data has been changed since the previous execution of the step using the input identical to the current input (column 7, lines 25-28);

Art Unit: 2165

and if the relevant data has not been changed, retrieving the output obtained in the previous execution of the step using the input identical to the current input (column 7, lines 57-59).

As per claim 26 Li is directed to determining whether the relevant data has been changed comprises:

retrieving a timestamp indicating a point of time of the previous execution (column 7, lines 53-55);

retrieving a transaction log of the database (column 7, line 59, wherein the log is the query holding the query information);

and determining, from the transaction log, whether transactions relative to the relevant data have occurred since the point of time indicated by the timestamp (column 7, lines 61-66).

As per claim 27 Li is directed to the operation further comprises:

if the step has not been executed using the input identical to the current input (column 7, lines 57-59, wherein the cache has no instance of stored transaction):

executing the step for the current input on the relevant data to obtain a result (column 6, lines 64-67; column 7, lines 1-3);

and storing the result to enable managing a next invocation of the step in which the step is passed input identical to the current input, in which case the

stored result is returned as output for the step without re-executing the step (column 7, lines 57-61).

As per claim 28 Li is directed to the operation further comprises:

if the relevant data has been changed since the previous execution of the step using the input identical to the current input (column 7, lines 62-67; column 8, lines 1-4):

executing the step for the current input on the relevant data to obtain a result (column 6, lines 64-67; column 7, lines 1-3);

and storing the result to enable managing a next invocation of the step in which the step is passed input identical to the current input, in which case the stored result is returned as output for the step without re-executing the step (column 7, lines 57-61).

As per claim 35 Ll is directed to a computer-implemented method of managing execution of a workflow that is repeatedly executed on data of a database, wherein the workflow is defined by plurality of steps, each being an executable function that operates on input from a previous step and produces output for subsequent step and each step having a defined input format and output format, the method comprising:

receiving current input for execution of a step of the workflow on relevant data of the database (column 6, lines 55-56; column 7, lines 55-61);

identifying the step as deterministic, whereby the step generates identical output for given input in repeated executions of the step on the relevant data (column 7, lines

Art Unit: 2165

56-57, wherein if the result check of the query is cached then it means that another query with identical input has been executed prior to this instance);

upon determining that the step had been previously executed using input identical to the current input, determining whether the relevant data has been changed since the previous execution of the step using the input identical to the current input (column 7, lines 25-24);

if the relevant data has been changed since the previous execution off the step using the input identical to the current input:

executing the step for the current input on the relevant data to obtain a result (column 7, lines 17-21); and

storing the result to enable managing a next invocation of the step in which the step is passed input identical to the current input, in which case the stored result is returned as output for the step without re-executing the step (column 7, lines 21-24); and

if the relevant data has not been changed since the previous execution of the step using the input identical to the current input:

returning output obtained in the previous execution of the step using the input identical to the current input without executing the step using the received current input (column 7, lines 52-61).



Art Unit: 2165

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 29-30, 32-34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Li (US 6,748,386 B1) in view of Crisan et al. (US 2003/0191769 A1).

As per claim 29 Li is directed to a computer system, comprising:

a database having data (column 3, lines 50-52);

the workflow execution manager being configured for:

receiving current input for execution of a step of the workflow on relevant data of the database, wherein the step has been previously executed on the relevant data using previous input identical to the current input and wherein the previous execution of the step produced previous output (column 6, lines 55-56; column 7, lines 55-61);

determining whether the step is deterministic, whereby the step generates identical output for given input in repeated executions of the step on the relevant data (column 7, lines 54-61, wherein the cached results would not be retrieved if the query wasn't the same);

and if the step is deterministic, returning the previous output produced during the previous execution of the step without re-executing the step (column 7, lines 55-61).

Li does not teach a workflow execution manager residing in memory for managing execution of a multi-step workflow that is repeatedly executed on the data of the database, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format.

Crisan et al. teaches a workflow execution manager residing in memory for managing execution of a multi-step workflow that is repeatedly executed on the data of the database, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format (Crisan et al., paragraph 0130, lines 4-15)

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the Li teachings of Crisan et al. to include providing an interface for specifying a single multi-analysis functional module used to execute the plurality of functional modules, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format because an output of one operation could be useful as an input for another.

As per claim 30 Li is directed to a computer system, comprising:  
a database having data (column 3, lines 50-52);

the workflow execution manager being configured for receiving current input for execution of a step of the workflow on relevant data of the database, wherein the step generates identical output for given input in repeated executions of the step on the relevant data (column 6, lines 55-56; column 7, lines 55-61);

and without executing the step using the current input, returning output obtained in a previous execution of the step using input identical to the current input (column 7, lines 57-61).

Li does not teach a workflow execution manager residing in memory for managing execution of a workflow that is repeatedly executed on the data of the database, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format.

Crisan et al. teaches a workflow execution manager residing in memory for managing execution of a workflow that is repeatedly executed on the data of the database, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format (Crisan et al., paragraph 0130, lines 4-15)

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the Li teachings of Crisan et al. to include providing an interface for specifying a single multi-analysis functional module used to execute the

Art Unit: 2165

plurality of functional modules, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format because an output of one operation could be useful as an input for another.

As per claim 32 Li is directed to a computer-implemented method of automatically executing a plurality of functional modules from within an application, comprising:

receiving current input for execution of at least one of the functional modules, wherein the at least one functional module has been previously executed using previous input identical to the current input (Li, column 6, lines 64-67; column 7, lines 1-7; column 7, lines 57-61);

determining whether the at least one functional module is deterministic, whereby the at least one functional module generates identical output for given input in repeated executions of the at least one functional module (Li, column 7, lines 54-61, wherein the cached results would not be retrieved if the query wasn't the same);

and if the at least one functional module is deterministic, returning previous output produced during the previous execution without re-executing the at least one functional module (Li, column 7, lines 55-61).

Li does not teach providing an interface for specifying a single multi-analysis functional module used to execute the plurality of functional modules, wherein the

workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format.

Crisan et al. teaches providing an interface for specifying a single multi-analysis functional module used to execute the plurality of functional modules, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format (Crisan et al., paragraph 0130, lines 4-15).

It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the Li teachings of Crisan et al. to include providing an interface for specifying a single multi-analysis functional module used to execute the plurality of functional modules, wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format because an output of one operation could be useful as an input for another.

As per claim 33 Li as modified is directed to comprising retrieving information regarding execution of the plurality of functional modules from a configuration file (Li, column 7, line 59, wherein "plurality of functional modules" means the "results" and "configuration file" means "cache").

As per claim 34 Li as modified is directed determining whether the at least one functional module is deterministic comprises examining information regarding the at least one functional module retrieved from the configuration file (Li, column 7, lines 61-66).

### ***Response to Arguments***

8. Applicant's arguments filed August 30, 2006 have been fully considered but they are not persuasive.

Applicant's argument to examiners objection to recitation of "whereby" in the claims is not deemed persuasive.

As per MPEP 2111.04 it states that "a whereby clause in a method claim is not given weight when it simply expresses the intended result of a process step positively recited". The statement following the "whereby" carries no weight since it only explains what is meant be deterministic. Should the applicant intend to have "the generation" step carry patentable weight, the claim language should be amended to direct functionality.

As to applicant's arguments that Li does not invoke a work flow the argument is deemed not persuasive.

In response to applicant's arguments, the recitation wherein the workflow is defined by plurality of steps, each step being an executable function that operates on input from a previous step and produces output for a subsequent step and each having a defined input format and output format has not been given patentable weight because the recitation occurs in the preamble. A preamble is generally not accorded any patentable weight where it merely recites the purpose of a process or the intended use of a structure, and where the body of the claim does not depend on the preamble for completeness but, instead, the process steps or structural limitations are able to stand alone. See *In re Hira*, 535 F.2d 67, 190 USPQ 15 (CCPA 1976) and *Kropa v. Robie*, 187 F.2d 150, 152, 88 USPQ 478, 481 (CCPA 1951).

As to applicant's arguments that Li does not teach "determining whether step is deterministic" is not deemed persuasive.

The claims do not describe how the invention makes such determination. The definition of deterministic is that the result is predictable or the same. Li makes that determination be checking the cache for prior results. If a result is found it means that the query with the same input has been executed before. Therefore the execution is halted and the result is retrieved from storage.

### ***Conclusion***

Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP

Art Unit: 2165

§ 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Tomasz Ponikiewski whose telephone number is (571)272-1721. The examiner can normally be reached on 8:00-4:30.

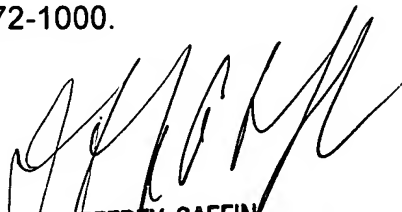
If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Jeffrey A. Gaffin can be reached on (571)272-4146. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.



Art Unit: 2165

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Tomasz Ponikiewski  
November 6, 2006



JEFFREY GAFFIN  
SUPERVISORY PATENT EXAMINER  
TECHNOLOGY CENTER 2100